## GQM APPROACH HAS EVOLVED TO INCLUDE MODELS

THE COMMENTS OF DAVE CARD about the Goal-Question-Metric approach ("What Makes for Effective Measurement?" Quality Time, Nov. 1993, pp. 94-95) refer to a very early version of the approach, referencing a 1984 paper (which was based upon a 1982 report of work done in 1979-80). I feel it is necessary to give a progress report.

Although it was originally used to define and evaluate goals for particular projects in particular environments, it is now used in a larger context, as the goal-setting step of the Quality Improvement Paradigm. The QIP is an evolutionary improvement approach tailored for software-development organizations. As such, we expanded GQM to include various types of experimental approaches to help clarify the design of QIP experiments.

The GQM has evolved into a model-based approach, using models of the process, product, and other forms of experience (see V.R. Basili and H.D. Rombach, "The TAME Project: Towards Improvement-Oriented Software Environments," *IEEE Transactions on Software Engineering*, June 1988, pp. 758-773). We use GQM to evaluate, characterize, predict, motivate, control, and focus improvement. Associated with the definition of a goal is a model-based interpretation.

This evolution of the GQM addresses many of the problems Card mentioned. But by 1988, the approach had gone even further. We developed a goal-generation template. Using the template, you analyze any objects (processes, products, other experience models), for a particular application's purpose (characterizing, evaluating, predicting, motivating, improving) with respect to various focuses (cost, correctness, defect removal, change, reliability, user friendliness, and so on) from different points of view (user, customer, manager, developer, corporation, and so on).

The GQM approach continues to evolve, based on the added requirements of the Software Engineering Laboratory's Experience Factory; the latest version is explained in "Software Modeling and Measurement: The Goal/Question/Metric Paradigm," (Tech. Report CS-TR-2956, UMIACS-TR-92-96, University of Maryland, 1992), which will appear in the proceedings of the CSR Workshop held in Amsterdam in October. It is also used at Hewlett-Packard and by the European AMI project in Europe.

It is true that many organizations prefer not to use the model-based version of the GQM and so do not benefit from the advantages that modeling provides. This may be because they do not have very good models of software experiences or because they believe the models to be implicit (the questions are used to extract information from the models).

Card also states that the GQM technique is

♦ *Not repeatable.* Not having models does not make the process nonrepeatable, because the goals, questions, and metrics are recorded. It is no less repeatable across different users than a design method. And, as with any approach, it can be misused.

♦ *Nonterminating.* The question of when to stop is a hard one. Surely models help, but good judgment is important. My own experience is that organizations have trouble stating goals at first, but once they get used to it, the problem becomes how to limit the number of goals. You stop when you have obtained sufficient information to assess the goal to the level of confidence necessary (partly determined by the point-of-view part of the template).

♦ *Not practical.* Card's concern that the approach is not practical because it may lead an organization to change the way it does business is unfounded. With the QIP, an organization first characterizes the way it does business by building models and baselines. If the goal is to understand that business, then surely it should not have to change in order to be measured. But if the goal is to improve, then change is desired.

*Vic Basili*
*University of Maryland*

## GQM PLUS HEURISTICS BETTER THAN BRAINSTORMING

THE PROBLEMS WITH GQM THAT Dave Card mentions can certainly occur, but I rarely encounter them. Perhaps it is because I have a few heuristics (not principles, unfortunately) that help.

For example, I urge people to restrict their goals and questions to a few important ones, perhaps 10. I also urge them to reuse goals and questions. (This advice works well in industrial settings. If you are engaged in a research project to discover new findings or new metrics, it probably does not help.)

I discard questions for which I cannot collect data. This seems obvious, but many people seem to want to find related data that doesn't quite answer the question and fool themselves into thinking that they answered it.

Perhaps most important, I emphasize that the purpose of the questions is two-fold: to satisfy the goals and to help define the measures and data to be collected. Accordingly, I discourage people from having more than one level of questions.

I have often observed that those who obtain good results from applying GQM find that some questions can be used to satisfy several goals. As a result, instead of having an exponential explosion of questions and measures, they get a modest increase from goals to questions to measures.

I quite agree that "engineering judgement and common sense" should be applied. In fact, they should not be considered supplements, but an integral part of the process. I would put it somewhat differently: Keep it simple!

Most of the strategy you define is worth having. Differentiating measurement goals from organizational goals is an integral part of GQM. When I teach or talk about GQM, I emphasize that the measurement goals are derived from organizational goals. For example, if a goal of the organization is to produce reliable software, then a measurement goal is to measure how reliable the software is that it produces. People who confuse measurement and organizational goals often cannot understand why they are asking certain questions. Those they ask are even more confused and cannot understand why they must provide data to answer those questions.

It is very important to continuously check the validity and accuracy of the data you collect. You must be sure you are collecting the data you think you are collecting and know what errors are in the data. (This is standard stuff in clinical psychology and other fields where attributes and performance of people